

REMARKS

The Examiner maintains the provisional obviousness-type double patenting rejection based on co-pending application serial number 09/944,114. Filed herewith is a Terminal Disclaimer that obviates this rejection. Withdrawal of the provisional obviousness-type double patenting rejection is requested.

Claims 1-7, 12, 14, 15, 17, 22-29, 34, 36-39, 44-51, 56, 58-61, and 66 stand rejected under 35 U.S.C. 103 as being unpatentable over Cheng in view of Landsman. This rejection is respectfully traversed.

The independent claims 1, 22, 23, 44, 45, and 66 describe detecting a tag in data received by a computer that indicates the existence of an updated version of a computer file being used by the computer. That tag detection triggers the computer to download an updated version of that computer file. This combination of features found in each independent claim is neither disclosed nor suggested by Cheng or Landsman. Moreover, the combination of Cheng and Landsman proposed by the Examiner is legally improper.

Cheng takes an entirely different approach to the problem of software updates. A service provider computer system stores in an update database information about software updates of various software vendors for particular software products. A user connects its client computer to the service provider computer to obtain a current version of the database (or at least some portion). Using this database information, the user manually selects one or more updates for installation. The selected updates are

downloaded from the software vendor computer system and installed on the client computer.

The Examiner relies on the embodiment in Cheng described at column 5, beginning at line 18, where software update information may be provided by the service provider to a client computer by email. But Cheng explicitly requires that such email notification be requested by the user. See column 5, line 21. Users must also indicate which software products they are interested in receiving when they request email notifications. Based on this user-triggered and user-provided information, the service provider detects when there is a software update that matches the user's interest and sends an email notifying the user of the software update. The email identifies the URL of where the software update is available. The user triggers the download by manually approving the installation of the software update. See column 5, lines 30-32.

As admitted by the Examiner, Cheng fails to disclose "[detecting] within data received by said computer a tag indicative of existence of an updated version of said computer file" (recited from claim 1). If a user indicates the user wants email notifications of updates, those email notifications are automatically sent. There is no need to detect a tag within that email—let alone a tag indicating the existence of an updated version of the computer file.

The Examiner also admits that Cheng fails to detect a tag to "trigger downloading from a predetermined source to provide said updated version of said computer file for use by said computer" (recited from claim 1). Indeed, any download in Cheng is triggered

specifically by manual user action and is not based upon tag detection. Of course, this makes complete sense because there is no tag detected in Cheng.

The Examiner tries to justify modifying Cheng's teachings by arguing that the use of an automated email to notify a user of an updated version "suggests a code process to automate the retrieval of files in a particular indication of latest file version analogous to tagging data for emphasis." See the bottom of page 6 of the Office Action. But Cheng's retrieval of files is not automated.

Cheng requires manual intervention by a user. First, the user must manually indicate that the user wants to receive email notifications of updates. Second, once an email update notice is received, the user must manually select and approve an update for installation before any downloading occurs. Cheng's process is not automated. Nor can it be reasonably said that sending an email notifying the user of a software update is analogous to embedding a tag in data. Cheng's email message must be detected, understood, and responded to by the human user. A tag is a simple message detected and responded to by a computer without any user intervention or decision making.

In an attempt to supply features admitted by the Examiner to be lacking from Cheng, the Examiner relies upon Landsman, which teaches providing interstitial web advertisements to a client computer. Particularly, the Examiner identifies advertising tag 40 which is embedded in a web page 35. A first component 42 of the advertising tag 40 is a script, which when executed by the client browser, functions as an applet agent to download advertisements to the client computer in a manner transparent to the user. The

second tag component 44 is the web address of the advertising management system server. The main goal of this two-component, ad tag is to "decouple" advertising content from the original web page content. See column 17, lines 22-24.

Although the client browser in Landsman detects the advertising tag 40 in the web page, that ad tag 40 does not indicate the "existence of an updated version of said computer file" (quoted from claim 1). Instead, the ad tag merely indicates a URL of an advertising management system. There is no updating of any computer file used by the client computer. Even the web page currently being viewed by the user is not updated. This makes sense because Landsman's main goal is to decouple advertising from the referring web page content.

Landsman's advertisement download is triggered by the executed script/applet 42. That ad content download from the ad management server is performed automatically. There is no download trigger based on detecting a tag in data received by the computer. Moreover, the download content is not: "said updated version of said computer file for use by said computer" (quoted from claim 1). Instead, the download is an advertisement—not any updated software.

Even Cheng and Landsman could be combined, for purposes of argument, their combination fails to disclose or suggest the combination of features recited in the independent claims. Neither reference teaches detecting in data received by the computer a tag that indicates the existence of an updated version of a computer file used by the computer. No tag is detected in Cheng because the email update notice is automatically

sent when the user manually requests those kinds of updates. The advertising tag in Landsman does not indicate the existence of an updated version of the computer file used by the computer. Neither reference discloses any computer file update triggering based on a detected tag that triggers downloading of the updated version of the computer file for use by the computer. Cheng's download is triggered by manual user action. Landsman's ad download is triggered by an executed script/applet and contains advertising content. Landsman's download does not update the version of any computer file used by the client computer.

The combination proposed by the Examiner is improperly based on hindsight. As the various Federal Circuit cases cited in the previous response explained, there must be some motivation or teaching in the prior art that would lead a person of ordinary skill to modify and combine Cheng and Landsman as proposed by the Examiner. The Examiner is not permitted to simply reconstruct the claim after the fact by selecting various features in various prior art references.

The Federal Circuit has confirmed that:

rejecting patents solely by finding prior art corollaries for the claimed elements would permit an Examiner to use the claimed invention itself as a blueprint for piecing together elements in the prior art to defeat the patentability of the claimed invention. Such an approach would be "an illogical and inappropriate process by which determine patentability."

Sensonics, Inc. v. Aerosonic Corp., 81 F.3d 1566, 1570 (Fed. Cir. 1996). But this is the very approach employed by the Examiner. The Examiner's reasoning at the bottom of page 6 of the Official Action attempts to gloss over Cheng's deficiencies using sweeping

prior art corollaries fashioned after-the-fact and after reading Applicant's claims.

Landsman does not even relate to software updates, but the Examiner draws elements from these very different types of prior art teachings without any evidence in Cheng or Landsman to do so.

In fact, both Cheng and Landsman *teach away* from the present invention. Cheng does so by advancing a software update approach, not based on detecting embedded tags, but rather based upon communications manually-initiated by a human user. Landsman explicitly teaches away by "totally decoupling referring web page content from its corresponding advertising content...[S]ince a referring web page merely incorporates an advertising tag totally devoid of advertising content, no changes whatsoever need to be made to that page." Column 13, lines 53-61. Thus, the data (web page) received by the client computer in Landsman, which includes the advertising tag, never changes. So how can that tag indicate the existence of a recently-updated version of a computer file? It can't.

Any attempt to modify Landsman closer to the independent claims would make Landsman inoperable for its primary "decoupling" purpose as articulated in the above-quoted text from column 13 and many places throughout Landsman. A proposed modification that renders a prior art reference inoperable for its intended purpose is inappropriate for an obviousness inquiry. *In re Fritch*, 972 F.2d 1260, 1265-1266 (Fed. Cir. 1992).

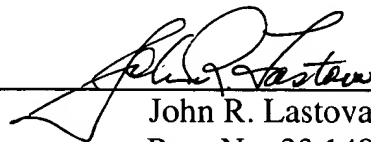
BARTON
Appl. No. 09/633,358
December 2, 2003

All of the rejections are based upon this faulty application of Cheng and Landsman. Accordingly, the application is now in condition for allowance. An early notice to that effect is earnestly solicited.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____



John R. Lastova
Reg. No. 33,149

JRL:at
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
Telephone: (703) 816-4000
Facsimile: (703) 816-4100